Supervised Machine Learning-based Wind Prediction to Enable Real-Time Flight Path Planning

Junghyun Kim*, Chao Zhang†, and Dimitri N. Mavris‡
Georgia Institute of Technology, Atlanta, Georgia, 30332, United States

Simon I. Briceno§
Jaunt Air Mobility, Atlanta, Georgia, 30307, United States

Many research groups have been committed to developing numerical models for weather forecasts. The models are currently used to predict weather patterns and trends in the aviation industry. In particular, pilots receive wind information predicted by the models and use the forecast to not only calculate how much fuel is needed for a flight but also optimize flight routes by seeking favorable winds. One potential issue is that the models provide relatively coarse wind information in both space and time, which potentially leads to inaccurate calculation of fuel consumption. This research aims to yield a continuous wind prediction model by combining a supervised learning algorithm with the Inverse Distance Weighting technique. Specifically, this research compares three different supervised learning algorithms that include Gaussian Process, Multi-Layer Perceptron, and Support Vector Machine to identify the most appropriate algorithm. The selected algorithm is then compared to a linear interpolation method that is widely used in current flight planning systems for obtaining continuous wind information. A case study is performed with the real Delta Airlines flight 1944 to evaluate the proposed methodology. The results show that 1) the Support Vector Machine provides a better wind prediction compared to the other models, 2) the supervised learning-based regression method performs better than the linear interpolation method in wind predictions, and 3) there are 16 seconds of difference between the real flight (12,117 seconds) and the simulated flight (12,101 seconds) for the cruise portion, indicating that the proposed methodology generates valid results as long as input wind data is provided accurately.

I. Introduction

While the number of flights in the United States (U.S.) has been dramatically decreased due to the COVID-19 pandemic [1], the fact that long-term forecasts for aviation indicate significant growth is still challenging. As aviation traffic rebounds, many airlines will be concerned about fuel consumption given that fuel accounts for up to 35 percent of airline operating costs [2]. Airlines could buy new aircraft and improve technologies (e.g., fuel-efficient engine); however, the required investment is presumably expensive for the airlines. As an operational solution, major airlines of the U.S. typically hire flight dispatchers to deal with weather-related flight operations. The flight dispatchers are responsible for coordinating all the preparations required to ensure a safe and efficient flight. Prior to departure time, they typically examine multiple information sources and integrate the sources to create an initial flight plan. More specifically, an initial flight plan is created based on the following procedures [3]: 1) enter all information (e.g., origin-destination) into a system, 2) find a recommended route from a list of company routes based on the information, 3) see if the recommended route penetrates areas of convective weather, 4) try to pick another route from a list of company routes if the recommended route penetrates any area, 5) create a new route based on multiple information sources if none of the company routes are available, and 6) release the flight plan. Figure 1 shows an example of the flight plan document.

In addition to avoiding areas of convective weather, it is important to create a flight plan by seeking favorable winds (i.e., riding a tailwind but avoiding headwinds) as winds can have a significant impact on an optimal flight route. For example, the previous Virgin Atlantic flight from Los Angeles to London on February 19th, 2019 shows the impact of

*Ph.D. Candidate, School of Computational Science and Engineering, and AIAA Student Member
†Assistant Professor, School of Computational Science and Engineering
‡Regent Professor, School of Aerospace Engineering, and AIAA Fellow
§Chief Commercial Officer, Jaunt Air Mobility, and AIAA Senior Member

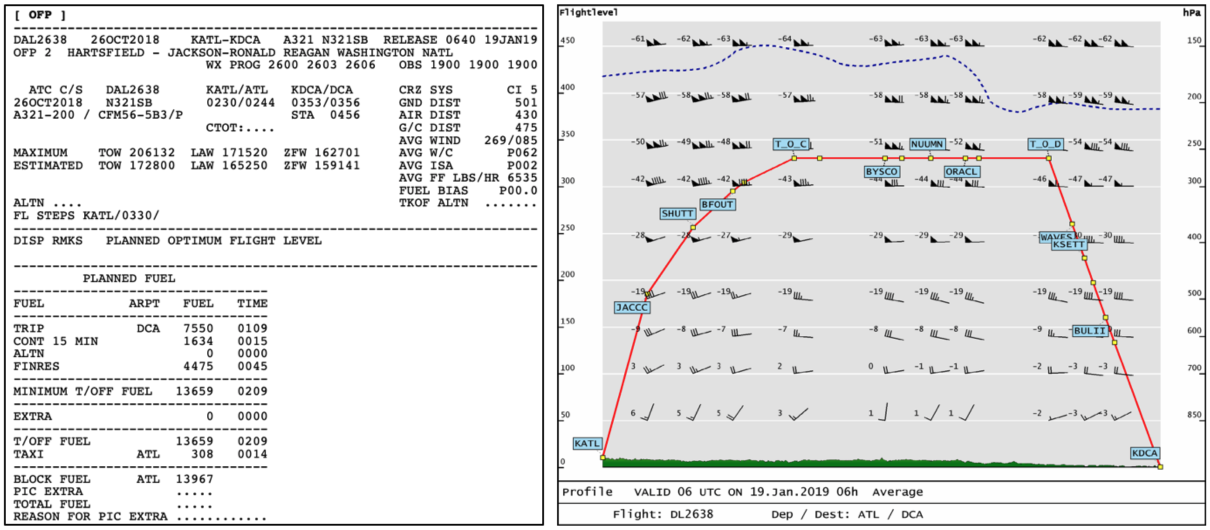


Fig. 1 Example of the flight plan generated by simBrief [4]

the wind on an optimal flight route. The flight achieved a record-breaking speed over central Pennsylvania due to the tailwinds so that the flight arrived 48 minutes early while the flight did not remain in the jet stream for long [5]. While the major airline flight dispatchers make significant efforts to not only avoid severe weather efficiently by continuously sharing up-to-date weather information but also minimize fuel consumption by proactively seeking favorable winds, it is worth mentioning that many small operators in the U.S. do not typically have flight dispatchers. The small operators generally either have a contract with a company that provides weather platforms [6] or use a flight path planning application. Figure 2 shows an example of a flight path planning application implemented in the flight deck.

Fig. 2 Traffic Aware Planner (TAP) installation in the cockpit [7]

One potential issue, however, is that many flight path planning applications do not raise a question about the resolution/quality of wind forecast data. This is significant because airlines encounter unexpected weather changes every day; thus, accurate wind information is imperative for airlines. Additional details about the potential issue will be discussed in the section of Problem Formulation. This research attempts to resolve the issue by proposing a hybrid approach that combines a supervised learning algorithm with the Inverse Distance Weighting (IDW) technique to yield a continuous wind prediction model in a more accurate manner. The outcome of this research could be used for the following use-cases: 1) improve the fidelity of the flight path optimization frameworks by providing better wind forecasts and 2) help current flight planning systems calculate fuel consumption in a more accurate manner.

II. Problem Formulation

Many research groups such as National Oceanic and Atmospheric Administration (NOAA) have been committed to developing numerical models for weather forecasts. The most well-known numerical weather models are tabulated in Table 1. The two representative weather forecast models are the Global Forecast System (GFS) developed by NOAA and the European Center for Medium-Range Weather Forecast (ECMWF) developed by European Centre called the American model and the European model, respectively. Figure 3 shows an example visualization of the GFS wind data plotted by the Panoply software [8].

Table 1 The most well-known numerical weather models

Model	Organization	Spatial Resolution	Temporal Resolution	Open Source	Region
MERRA-2	NASA	27 km	3 hours	Yes	Global
NAM	NOAA	12 km	3 hours	Yes	US
RAP	NOAA	13 km	1 hour	Yes	US
GFS	NOAA	13 km	1 hour	Yes	Global
HRRR	NOAA	3 km	1 hour	Yes	US
ECMWF	CoE	9 km	1 hour	No	Global

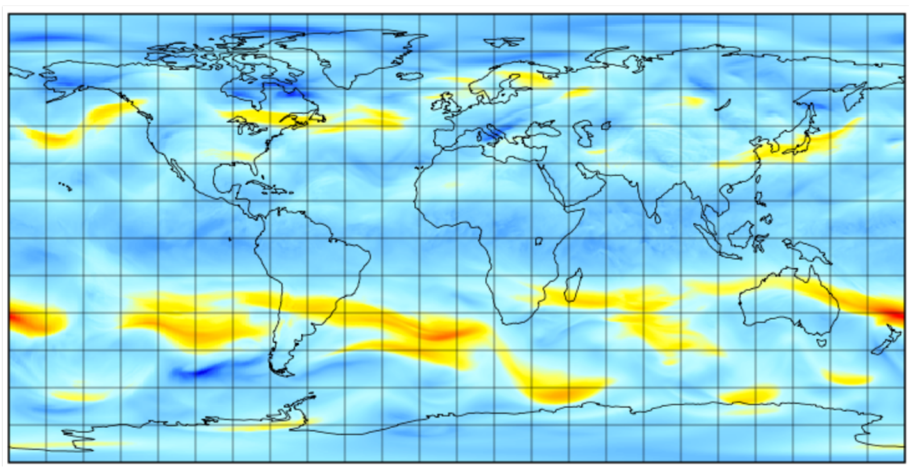


Fig. 3 GFS eastward wind visualization by Panoply

With the development of weather forecast models, many research groups have been able to predict weather patterns and trends. In particular, the wind is one of the parameters best predicted by the models as it can be directly estimated from the physical equations (e.g., Navier-Stokes equations) [9]. For this reason, pilots typically receive wind information predicted by the models and use the forecast to not only calculate how much fuel is needed for a flight but also optimize flight routes by seeking favorable winds. One potential issue, however, is that the models provide relatively coarse (i.e., discrete) wind information in both space and time, potentially leading to inaccurate calculation of fuel consumption. Figure 4 notionally illustrates the reason why it is imperative to create a reliable model that provides continuous wind information with respect to four-dimensional (i.e., timestamp, altitude, latitude, and longitude) flight trajectory.

The most common approach for obtaining continuous wind information in current flight planning systems is via a linear interpolation method. For example, the Traffic Aware Planner (TAP) employs a linear interpolation method to provide continuous Rapid Refresh (RAP) wind data to the in-flight re-planning system [10]. The Aviation Environmental Design Tool (AEDT) uses a linear interpolation method with the Modern-Era Retrospective analysis for Research and Applications-2 (MERRA-2) wind data to calculate fuel consumption in the simulation environment [11]. The Real-time Trajectory OPtimization (RTOP) tool utilizes a linear interpolation method with the High Resolution Rapid Refresh (HRRR) wind data to calculate total travel time [12]. The linear interpolation method may be appropriate for obtaining continuous wind information at cruising altitudes given that winds do not change dramatically over time at

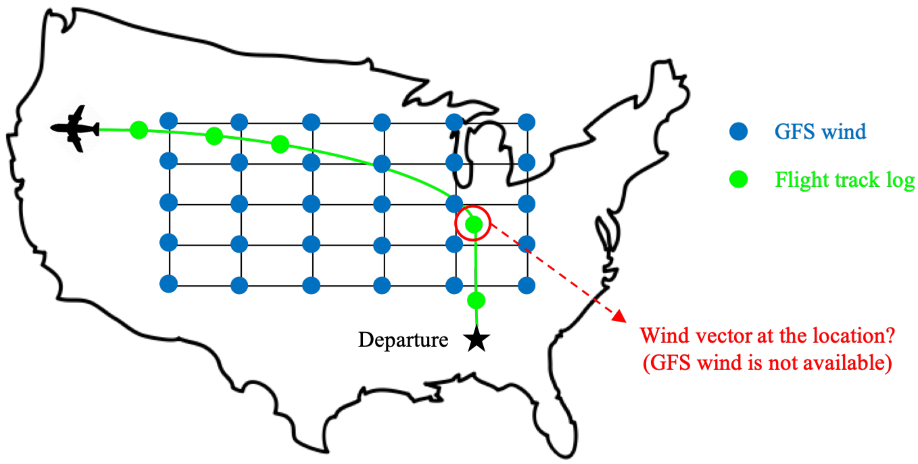


Fig. 4 Notional sketch of the necessity for continuous wind information

these altitudes. However, the linear interpolation method may have some limitations in predicting wind patterns where non-linear behavior is dominant. Based on these observations, the following research question can be constructed:

Research question: How can we provide continuous wind information to a real-time flight path planning framework in a more accurate manner?

Recently, with the advent of Artificial Intelligence (AI), many machine learning algorithms have been widely used with a data-driven approach to enhancing the level of understanding of wind phenomena. For example, Ashish Kapoor et al. [13] developed a probabilistic graphical model with a data-driven approach to estimate wind and aircraft true airspeed. M.A. Mohandes et al. [14] compared two different machine learning algorithms for wind speed prediction. Ronay Ak et al. [15] combined extreme learning machines with the nearest neighbors approach for short-term wind speed time-series prediction. Aditya Grover et al. [16] proposed a hybrid approach that combines discriminatively trained predictive models with a deep neural network for weather forecasting. While these methodologies [13–16] have proved their capabilities in a certain application, they may limit their usefulness for real-time analyses due to high computational costs (e.g., constructing deep learning networks). Given that the ultimate goal of this research is to yield a continuous wind prediction model for a real-time flight path planning framework, in this paper, we propose a supervised learning-based wind prediction methodology that is feasible to provide wind data along a flight trajectory in real-time. Additional details will be discussed in the Proposed Methodology section.

III. Methodology

A. Data acquisition

After careful consideration of the most well-known weather products summarized in Table 1, the GFS was selected as primary data sources for the following reasons: 1) the GFS is commercially available so that users can easily connect to the server and download the data, 2) the GFS covers the entire globe in addition to U.S. territory, which potentially shows a possibility in which this research can be extended to an international flight analysis, and 3) NOAA recently made several significant technological improvements on the GFS [17]. The GFS is a global weather forecast model produced by National Centers for Environmental Prediction (NCEP) and can be downloaded from the official website [18]. The GFS is updated every six hours (i.e., 00, 06, 12, 18 UTC) and provides a set of detailed weather-related properties such as temperature, pressure, wind speed/direction, and relative humidity against longitude, latitude, altitude, and timestamp. The GFS has an approximate horizontal resolution of 13 kilometers and divides the Earth into 64 layers vertically.

B. Data preparation

We implemented a data pipeline (i.e., (*DATA.py*) to retrieve the GFS weather data from the NOAA data server and to automatically go through data pre-processing steps. Among various GFS weather parameters (e.g., temperature), we primarily concentrated on wind speed and direction data in this research. Figure 5 shows an example visualization of the GFS wind at a specific time and altitude generated by the Python code.

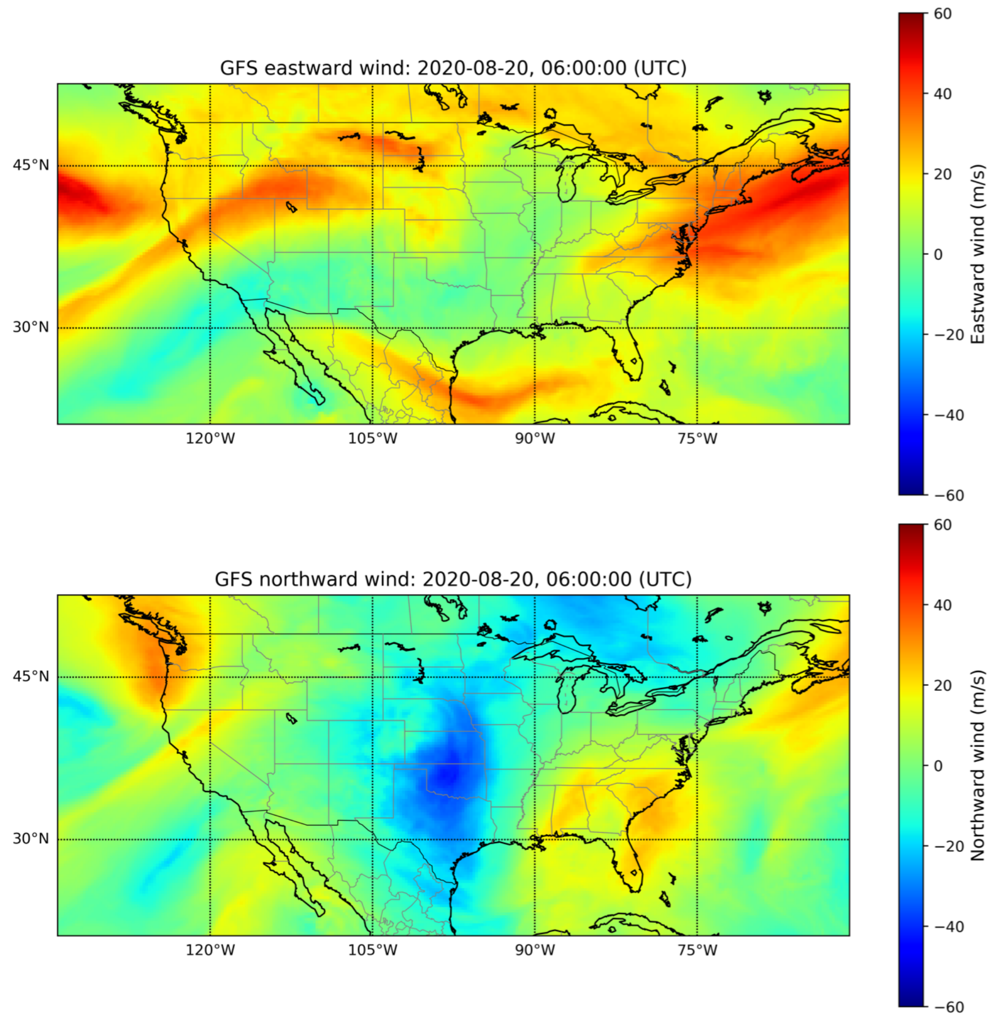


Fig. 5 GFS wind visualization at 2020-08-20 06:00 UTC (Altitude = 250 hPa)

The data pre-processing steps implemented in the Python code are as follows: 1) decode the original GFS data using the Pygrib [19] library, 2) reconstruct the wind data from a two-dimensional table to an input-output column for a learning process, and 3) decompose the data into training (80%) and validation (20%) datasets. Here, the validation datasets are randomly selected.

C. Supervised learning-based wind prediction

In this paper, a hybrid approach that combines a supervised learning algorithm with the IDW technique is presented to yield a continuous wind prediction model. The hybrid approach consists of two parts that include 1) spatial regression and 2) temporal interpolation. Additional details will be discussed below.

1. Spatial regression

First, we implemented the Multi-Layer Perceptron (MLP) to create a non-linear regression model of the GFS wind data with the aim of finding the best weight parameters to minimize errors between predicted and target values by adopting an iterative procedure as shown in Figure 6. To find the best weight parameters in the model, we implemented the Adam algorithm [20] which is an extended version of the stochastic gradient descent method. The MLP-based wind regression model entails the following fully-connected layers as shown in Figure 7: 1) an input layer to receive the GFS wind data, 2) an output layer with the linear activation function that makes a prediction, and 3) two hidden layers with the Sigmoid function that are the true computational engine for the regression.

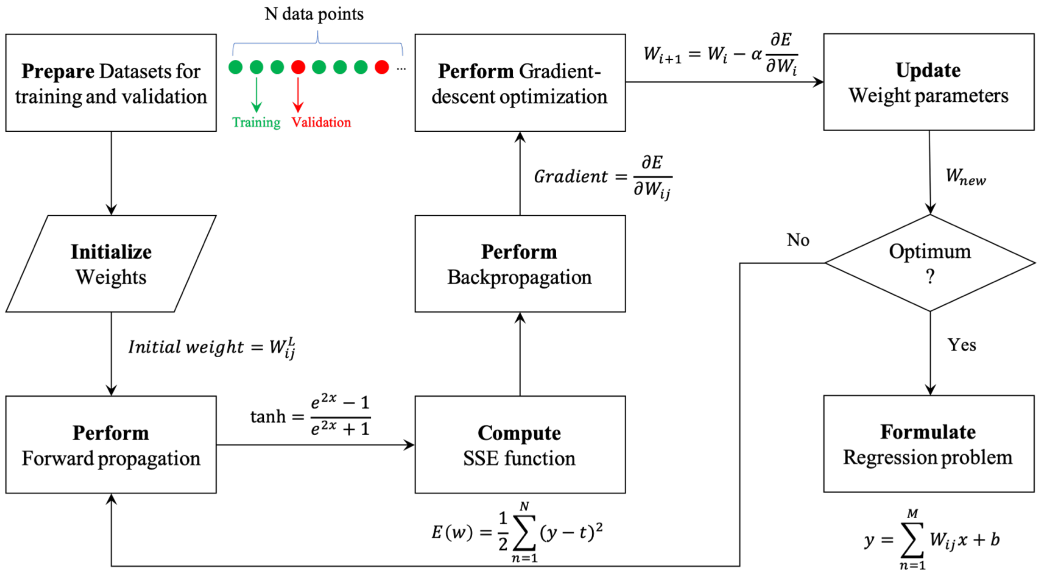


Fig. 6 Flowchart of the MLP-based wind regression modeling process

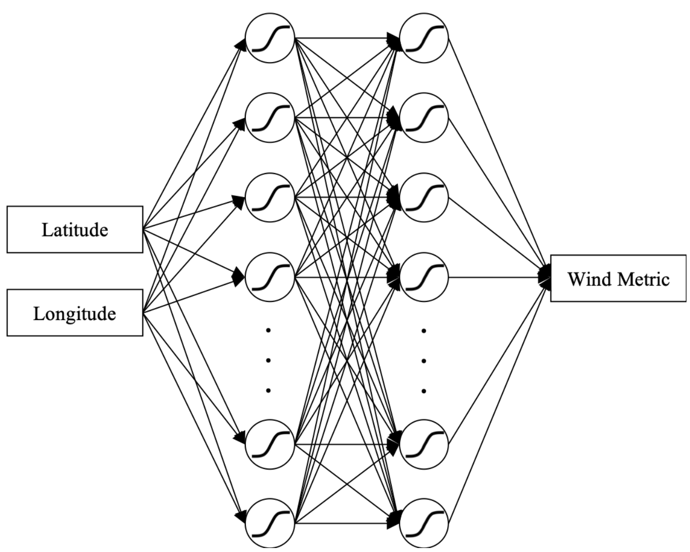


Fig. 7 Diagram of the MLP-based wind regression model structure (50 hidden nodes)

Second, we implemented the Support Vector Regression (SVR) [21] to obtain continuous values of the GFS eastward and northward wind in an entire two-dimensional space. Since the GFS wind data is not linearly separable in the original feature space, we evaluated a few valid (i.e., symmetric and positive semi-definite) kernel functions to transform the feature space. We were determined to choose the Radial Basis Function (RBF) kernel as we observed that it performed better than the other kernel functions. The choice of the RBF kernel function naturally led us to consider two hyper-parameters of the algorithm that are the penalty parameter C and the RBF-related parameter r as they had greatly impacted the model performance. We controlled the penalty C parameter to balance between correct and incorrect data points in the objective function of the optimization problem. In addition, we tuned the r parameter to ensure that the algorithm was not too constrained and affected by an over-fitting issue. In particular, we implemented a grid search method as notionally shown in Figure 8 to find out the best values for the hyper-parameters. The effective Support Vector Machine (SVM) algorithm, especially for given GFS wind data, was finally determined by the choice of hyper-parameters.

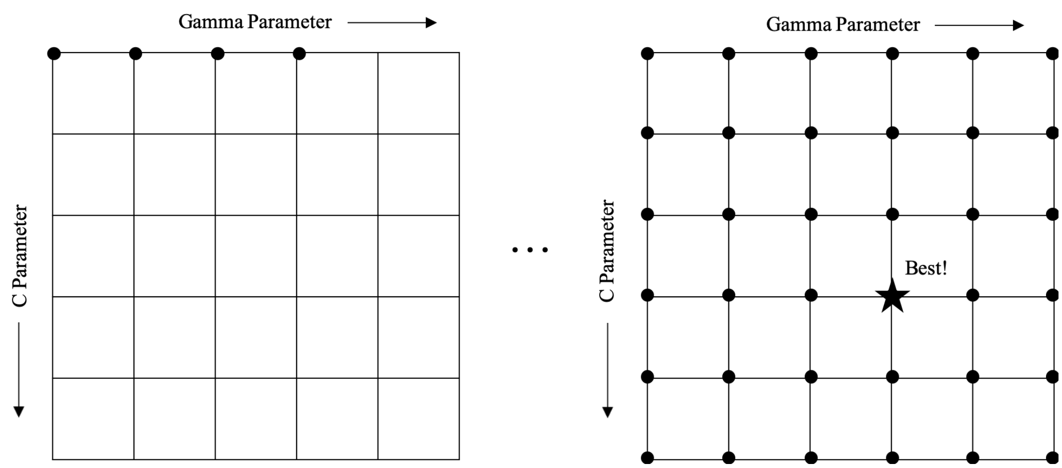


Fig. 8 Notional sketch of grid search approach for choosing the SVR hyper-parameters

Third, we implemented the Gaussian Process (GP) to create a non-linear probabilistic regression model of the GFS wind data. Given N input and output pairs, we utilized the squared-exponential kernel function, which is defined as $k(x_i, x_j) = \theta_0 \exp(-\frac{\theta_1}{2}\|x_i - x_j\|^2)$, where the hyper-parameters of the kernel were optimized during the GP regression process by maximizing the log-marginal-likelihood of the outputs.

2. Temporal interpolation

As the GFS is an hourly updated, we need a way to estimate the wind speed and direction at any specific time within an hour. We implemented the IDW technique to create a temporal interpolation on the GFS wind data. Figure 9 depicts how to perform the IDW with the basic form defined by Shepard [22]. The IDW method basically assumes that closer values are more related than further values.

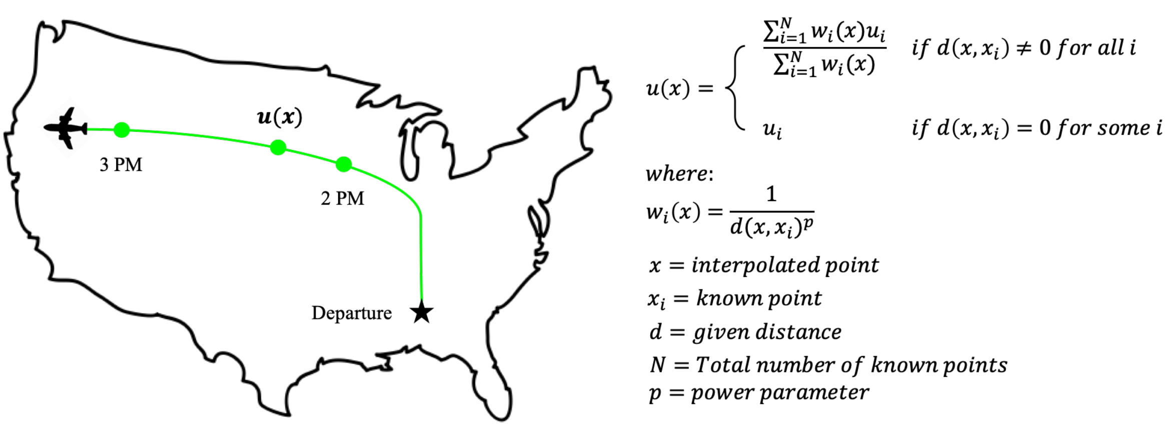


Fig. 9 IDW method for GFS wind temporal interpolation

IV. Results and Discussion

A. Supervised learning-based wind regression model evaluation

To identify the most appropriate algorithm for spatial regression, we computed the coefficient of determination (i.e., R-squared) for three different models. The R-squared is an estimate of the proportion in the variation around the mean that can be attributed to the model rather than the random error (i.e., how well the model predictions adhere to reality). As can be seen in Table 2, the R-squared results show that the SVM provides a better prediction compared to the other models. However, it is important to note that a high value of the R-squared does not imply that the model is accurate; but it only provides an indication as to whether or not the next step of checking model performance can be progressed.

Table 2 Supervised learning-based wind prediction model evaluation

	RMSE (m/s)		R-squared	
Model	Eastward Wind	Northward Wind	Eastward Wind	Northward Wind
SVM	1.39	1.28	0.99	0.99
MLP	3.65	3.94	0.95	0.94
GP	1.88	1.83	0.97	0.98

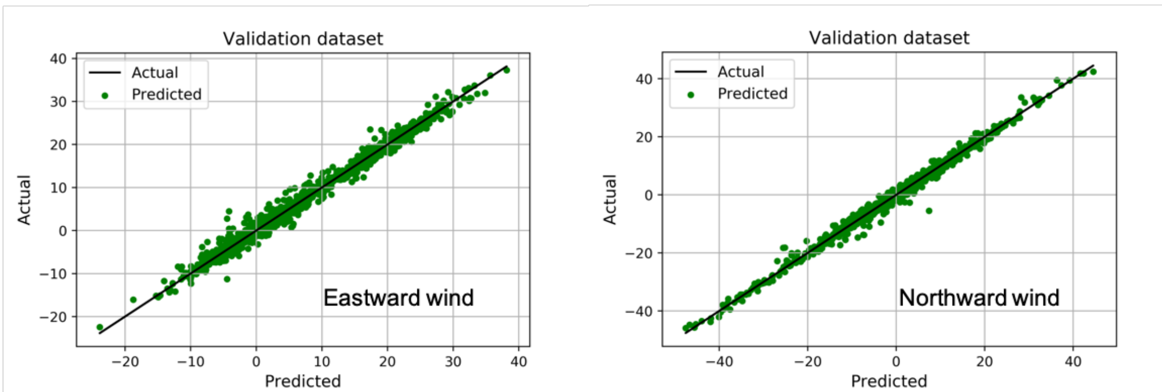
As the next step of model evaluation, we calculated the Root Mean Square Error (RMSE) to estimate the standard deviation of the random error (i.e., how residuals are spread out). In particular, we investigated two different types of RMSE distributions that involve 1) Model Fit Error (MFE) that represents how well the machine learning model fits the wind data points and 2) Model Representation Error (MRE) that represents how well the machine learning model predicts the actual response. Since the MFE (i.e., training error) was not sufficient for the model evaluation process, we specifically focused on the MRE (i.e., validation error) to ensure the predictive capability of the supervised learning-based wind regression models. The MRE results are summarized in Table 2. As expected, the SVM model performed better than the other models. Figure 10 shows the results of the MREs for the supervised learning-based wind regression models.

B. Supervised learning-based regression vs. Linear interpolation

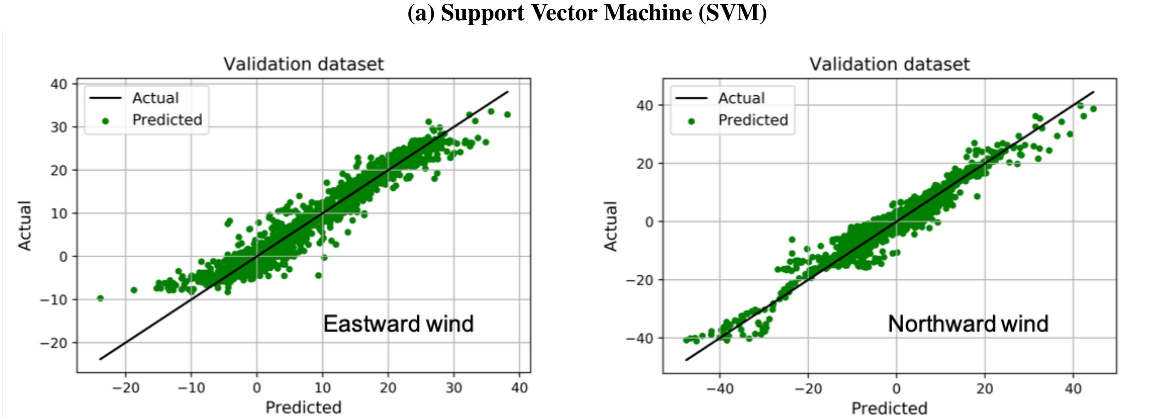
We randomly sampled 50 points (i.e., testing points) as shown in Figure 11b to evaluate the performance of two approaches, namely supervised learning-based regression and linear interpolation, by comparing with the GFS wind value at the testing points. We ended up obtaining three different values at the testing points: 1) GFS wind value, 2) wind value estimated by the supervised learning-based regression model, and 3) wind value estimated by the linear interpolation method. Figure 11a illustrates how the comparison is performed. In Step 1, the red dots notionally represent the validation points and the blue dots represent the training points for the supervised learning-based regression method, respectively. The green dots refer to the testing points used for the comparison (i.e., supervised learning-based regression vs. linear interpolation). In Step 2, the black dots refer to the points used for a linear interpolation method and the green dot (i.e., P_1) is the point predicted by the linear interpolation method. Table 3 shows the results of the RMSE for three different machine learning models and the linear interpolation model, indicating that all the supervised learning-based regression methods provided a better wind prediction compared to the linear interpolation method for the validation points.

Table 3 Supervised learning-based regression vs. Linear interpolation

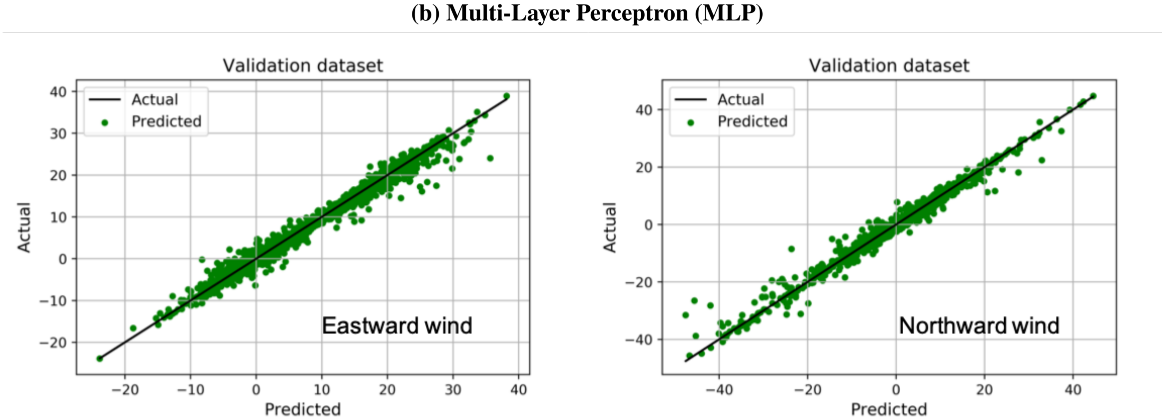
Method	Eastward wind RMSE (m/s)	Northward wind RMSE (m/s)
Support Vector Machine	0.79	0.87
Multi-Layer Perceptron	1.14	1.36
Gaussian Process	0.91	1.28
Linear Interpolation	1.21	1.45



(a) Support Vector Machine (SVM)

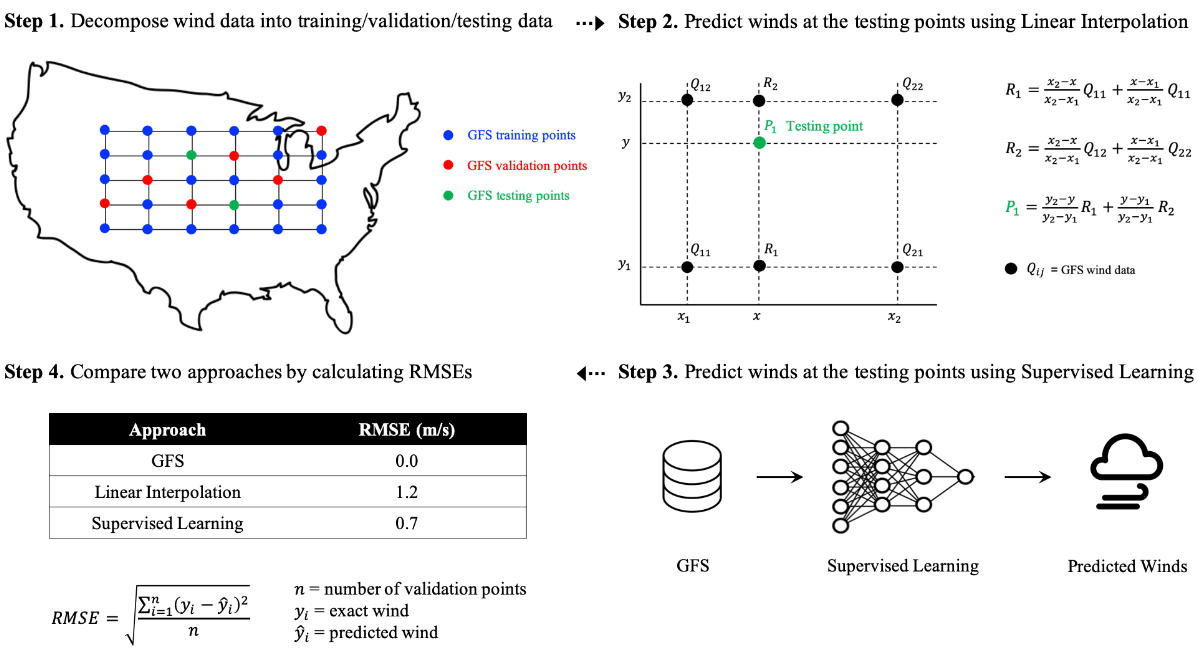


(b) Multi-Layer Perceptron (MLP)



(c) Gaussian Process (GP)

Fig. 10 Actual vs. Predicted plots of the supervised learning-based wind regression models

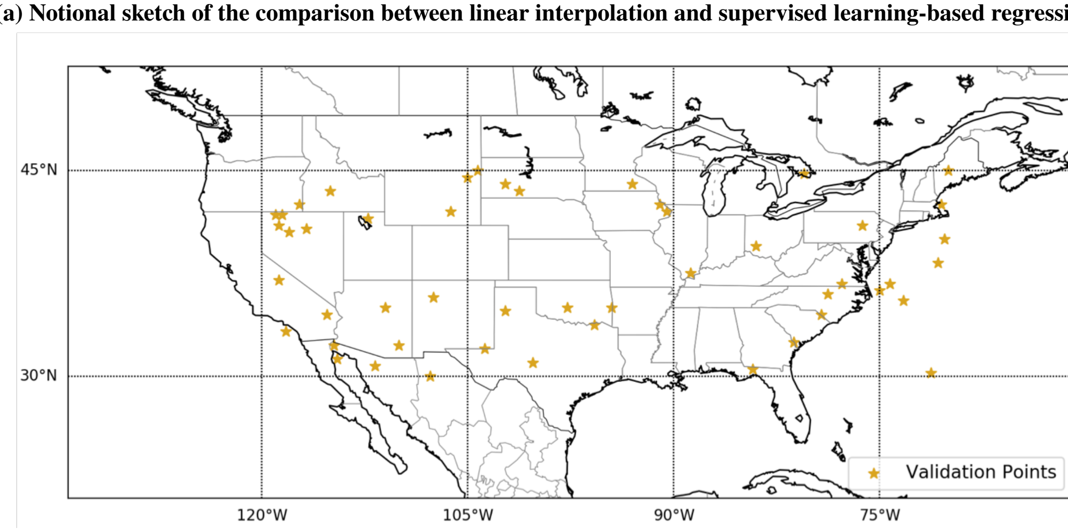


Approach	RMSE (m/s)
GFS	0.0
Linear Interpolation	1.2
Supervised Learning	0.7

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}}$$

n = number of validation points
y_i = exact wind
$\hat{y}_i$ = predicted wind

(a) Notional sketch of the comparison between linear interpolation and supervised learning-based regression



(b) Validation points for the comparison between linear interpolation and supervised learning-based regression

Fig. 11 Comparison between linear interpolation and supervised learning-based regression

C. Case study: Delta Airlines 1944 flight

To evaluate the proposed methodology, we conducted a case study that focused on a fair comparison between the real flight and simulation. Figure 12 shows the trajectory of the previous Delta Airlines Flight 1944 used for the comparison. It should be noted that we concentrated only on the cruise phase because the cruise portion was the longest segment of the selected flight.

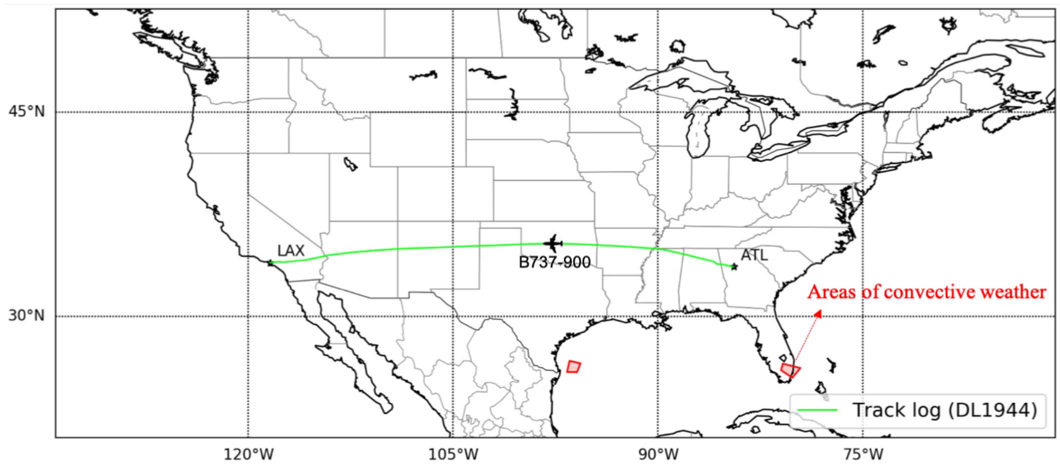


Fig. 12 DL1944 flight path visualization

Specifically, we went through the following steps: 1) collect the previous Delta Airlines flight information (i.e., altitude, ground speed, latitude, longitude, and time) and the corresponding GFS weather data, 2) collect true airspeed information for the selected flight from Air Navigation Service Provider (ANSP), 3) constrain the simulated flight to follow the real flight trajectory, and 4) calculate the duration of the flight. For the steps 3 and 4, we developed a Python code (*AIRCRAFT.py*) to estimate travel time for the selected flight route. A high-level summary of the implementation is presented in Algorithm 1.

Algorithm 1 Travel Time Function in *AIRCRAFT.py*

```
Specify the user-defined maximum distance (d)
function ESTIMATED TRAVEL TIME(Origin, Destination, True Airspeed, Time)
    Calculate travel distance (D) from an origin and destination using the Haversine formula
    if D < d then
        Compute aircraft initial bearing
        Estimate wind values using the proposed methodology
        Calculate aircraft ground speed (G)
        travel time = D / G
    else if D > d then
        Discretize the trajectory into N segments
        for segment = 1, 2, . . . , N do
            Calculate travel distance for the segment using the Haversine formula (D_s)
            Compute aircraft initial bearing at the start point of the segment
            Estimate wind values at the start point of the segment using the proposed methodology
            Calculate aircraft ground speed at the start point of the segment (G_s)
            travel time for the segment = D_s / G_s
        end for
        Sum travel time for all the segments
    end if
end function
```

Comparing the duration of the cruise segment, there were 16 seconds of difference between the real flight and the simulated flight as shown in Table 4. This indicates that the proposed methodology generates valid results as long as input data (e.g., wind and aircraft performance) is provided accurately.

Table 4 Real flight (DL1944) vs. Simulation

Flight Mode	DL1944	Simulation
Taxi out	00:19:00	00:00:00
Climb	00:21:14	00:00:00
Cruise	03:21:57	03:21:41
Descend	00:22:46	00:00:00
Taxi in	00:06:00	00:00:00

V. Conclusion

This research aimed to address potential issues of the current flight planning systems that utilize wind information predicted by the numerical models. A hybrid approach that combined supervised learning algorithms with the Inverse Distance Weighting technique was presented to yield a continuous wind prediction model. Three different supervised learning algorithms were compared to identify the most appropriate model in wind predictions. As a result, the Support Vector Machine provided a better wind prediction compared to the other models (i.e., Gaussian Process and Multi-Layer Perceptron) at the validation points. The Support Vector Machine was then compared to a linear interpolation method at the testing points. The results showed that the Support Vector Machine performed better than the linear interpolation method in wind predictions. A case study was performed with the real Delta Airlines flight 1944 to evaluate the proposed methodology. The algorithm (i.e., *AIRCRAFT.py*) constrained the simulated flight to follow the real flight trajectory especially for the cruise segment, and calculated the duration of the simulated flight. The results showed that there were 16 seconds of difference between the real flight (12,117 seconds) and the simulated flight (12,101 seconds), indicating that the proposed methodology generates valid results as long as input wind data is provided accurately. The outcome of this research could be integrated into the current real-time flight path planning tools that mostly employ a linear interpolation method to account for continuous wind information. This potentially leads to the improvement of the fidelity of the real-time flight path planning frameworks.

VI. Future Work

This paper primarily accounted for the supervised learning-based wind regression modeling problem. In the end, the author aims to develop an automated framework that performs in-flight re-planning in a more accurate and frequent manner. Since the short-term convective SIGMET modeling problem was already presented in the paper [23], future works will incorporate the outcome of this research into the real-time flight path optimization framework as depicted in Figure 13.

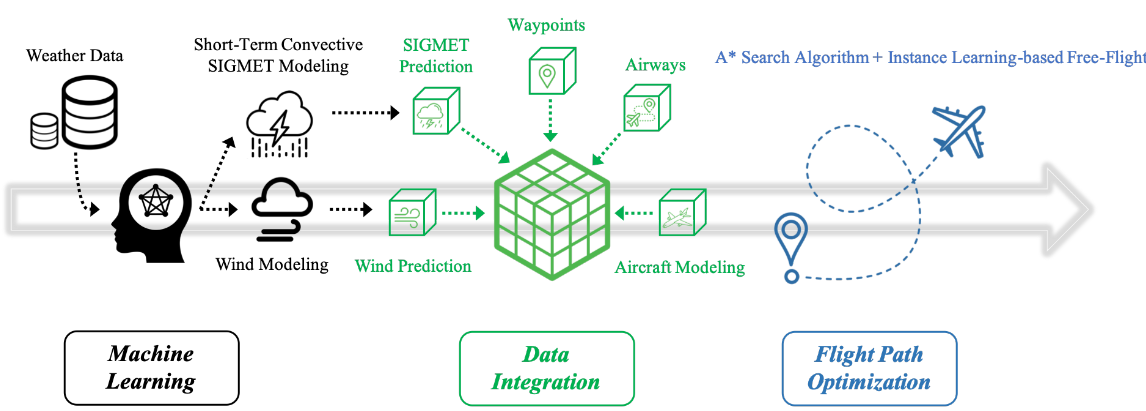


Fig. 13 Research overview

References

[1] International Civil Aviation Organization, "COVID-19 Air Traffic Dashboard," `https://www.icao.int/sustainability/Pages/COVID-19-Air-Traffic-Dashboard.aspx`, Retrieved: 2020-11-30.

[2] U.S. Department of Transportation, "What the Cost of Airline Fuel Means to You," `https://www.transportation.gov/administrations/assistant-secretary-research-and-technology/what-cost-airline-fuel-means-you`, Accessed: 2020-11-30.

[3] Dorneich, M. C., Olofinboba, O., Pratt, S., and Feyereisen, T., "Integration of weather information into the dispatcher pre-flight route selection process," *Proceedings. The 21st Digital Avionics Systems Conference*, Vol. 2, IEEE, 2002, pp. 7A1–7A1. https://doi.org/10.1109/dasc.2002.1052913.

[4] Derek Mayer, "simBrief," https://www.simbrief.com/home/, Retrieved: 2020-11-30.

[5] The Washington Post, "Flight reaches 801 mph as a furious jet stream packs record-breaking speeds," https://www.washingtonpost.com/weather/2019/02/19/flight-reaches-mph-furious-jet-stream-packs-record-breaking-speeds/, Retrieved: 2020-11-30.

[6] The Weather Company, "Advancing weather science to help improve forecasts for people and businesses everywhere," https://www.ibm.com/weather, Retrieved: 2020-11-30.

[7] Lewis, T. A., Burke, K. A., Underwood, M. C., and Wing, D. J., "Weather Design Considerations for the TASAR Traffic Aware Planner," *AIAA Aviation 2019 Forum*, 2019, p. 3616. https://doi.org/10.2514/6.2019-3616.

[8] National Aeronautics and Space Administration, "Panoply," https://www.giss.nasa.gov/tools/panoply/, Retrieved: 2020-11-30.

[9] Rozas-Larraondo, P., Inza, I., and Lozano, J. A., "A method for wind speed forecasting in airports based on nonparametric regression," *Weather and Forecasting*, Vol. 29, No. 6, 2014, pp. 1332–1342. https://doi.org/10.1175/waf-d-14-00006.1.

[10] Lewis, T. A., Burke, K. A., Underwood, M. C., and Wing, D. J., "Weather Design Considerations for the TASAR Traffic Aware Planner," *AIAA Aviation 2019 Forum*, 2019, p. 3616. https://doi.org/10.2514/6.2019-3616.

[11] Ahearn, M., Boeker, E., Gorshkov, S., Hansen, A., Hwang, S., Koopmann, J., Malwitz, A., Noel, G., Reherman, C. N., Senzig, D. A., et al., "Aviation Environmental Design Tool (AEDT) Technical Manual: Version 2c," Tech. rep., United States. Federal Aviation Administration. Office of Environment and Energy, 2016.

[12] Ramee, C., Kim, J., Deguisgnet, M., Justin, C., Briceno, S., and Mavris, D. N., "Aircraft Flight Plan Optimization with Dynamic Weather and Airspace Constraints," *International Conference for Research in Air Transportation (ICRAT)*, 2020.

[13] Kapoor, A., Horvitz, Z., Laube, S., and Horvitz, E., "Airplanes aloft as a sensor network for wind forecasting," *IPSN-14 Proceedings of the 13th International Symposium on Information Processing in Sensor Networks*, IEEE, 2014, pp. 25–33. https://doi.org/10.1109/ipsn.2014.6846738.

[14] Mohandes, M. A., Halawani, T. O., Rehman, S., and Hussain, A. A., "Support vector machines for wind speed prediction," *Renewable energy*, Vol. 29, No. 6, 2004, pp. 939–947. https://doi.org/10.1016/j.renene.2003.11.009.

[15] Ak, R., Fink, O., and Zio, E., "Two machine learning approaches for short-term wind speed time-series prediction," *IEEE transactions on neural networks and learning systems*, Vol. 27, No. 8, 2015, pp. 1734–1747. https://doi.org/10.1109/tnnls.2015.2418739.

[16] Grover, A., Kapoor, A., and Horvitz, E., "A deep hybrid model for weather forecasting," *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2015, pp. 379–386.

[17] National Oceanic Atmospheric Administration, "NOAA upgrades the U.S. global weather forecast model," https://www.noaa.gov/media-release/noaa-upgrades-us-global-weather-forecast-model, Retrieved: 2020-11-30.

[18] National Oceanic Atmospheric Administration, "Global Forecast System (GFS)," https://www.ncdc.noaa.gov/data-access/model-data/model-datasets/global-forcast-system-gfs, Retrieved: 2020-11-30.

[19] Pygrib, "Python module for reading and writing GRIB files," https://jswhit.github.io/pygrib/docs/index.html, Retrieved: 2020-11-30.

[20] Kingma, D. P., and Ba, J., "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[21] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al., "Scikit-learn: Machine learning in Python," *the Journal of machine Learning research*, Vol. 12, 2011, pp. 2825–2830.

[22] Shepard, D., "A two-dimensional interpolation function for irregularly-spaced data," *Proceedings of the 1968 23rd ACM national conference*, 1968, pp. 517–524. https://doi.org/10.1145/800186.810616.

[23] Kim, J., Briceno, S., Justin, C., and Mavris, D., "A Data-Driven Approach using Machine Learning to Enable Real-Time Flight Path Planning," *AIAA Aviation 2020 Forum*, 2020, p. 2873. https://doi.org/10.2514/6.2020-2873.